MULTISCALE MODEL. SIMUL.
Vol. ?, No. ?, pp. ?
© 2009 Society for Industrial and Applied Mathematics

P-SPLINES USING DERIVATIVE INFORMATION *

CHRISTOPHER P. CALDERON †, JOSUE G. MARTINEZ§, RAYMOND J. CARROLL§, AND DANNY C. SORENSEN †

Abstract. Time series associated with single-molecule experiments and/or simulations contain a wealth of multiscale information about complex biomolecular systems. We demonstrate how a collection of Penalized splines (P-Splines) can be useful in quantitatively summarizing such data. In this work, functions estimated using P-splines are associated with stochastic differential equations (SDEs). It is shown how quantities estimated in a single SDE summarize fast-scale phenomena, whereas variation between curves associated with different SDEs partially reflects noise induced by motion evolving on a slower time scale. P-splines assist in "semi-parametrically" estimating nonlinear SDEs in situations where a time dependent external force is applied to a single-molecule system. The P-splines introduced simultaneously use function and derivative scatterplot information to refine curve estimates. We refer to the approach as the PuDI (P-Splines using Derivative Information) method. It is shown how generalized least squares ideas fit seamlessly into the PuDI method. Applications demonstrating how utilizing uncertainty information/approximations along with generalized least squares techniques improve PuDI fits are presented. Although the primary application here is in estimating nonlinear SDEs, the PuDI method is applicable to situations where both unbiased function and derivative estimates are available.

1. Introduction. Our interest is modeling time series associated with single-molecule simulations/experiments [1–3]. We demonstrate how information in such time series can be summarized into scatterplot data [2, 5] and how a new method introduced here, the P-splines using Derivative Information (PuDI) method, can be used to gain better quantitative understanding of these time series containing information about multiple time scales. The motivation for quantitatively modeling the stochastic dynamics stems from recent technological advances in single-molecule physics. These advances have made it possible to manipulate individual macromolecules and measure various kinetic and thermodynamic properties associated with complex molecules, such as proteins and nucleic acids, at nanoscale resolution without artifacts associated with bulk measurements obscuring results. For example, a high resolution atomic force microscope (AFM) was recently used to measure the force time series associated with repeatedly unfolding and re-folding a single protein. The study demonstrated that modifying the chemical environment via ligand concentration alters the protein folding kinetics [4]. Information of this sort can provide researchers with a new level of fundamental understanding and can also be exploited in novel nanotechnology/molecular medicine applications. However the complexity of the underlying system and the stochastic dynamics inherent in small scales rarely permit a simple parametric model to accurately approximate the global stochastic dynamics. Another complication stems from the fact that the external forces introduced into the system typically result in non-stationary time series. We demonstrate how the PuDI method can help in addressing these complications. In this article, the focus in on a benchmark all-atom molecular dynamics simulation of gramicidin A [3], but the method outlined

*Received by the editors August 13, 2009; CPC would like to thank Lorant Janosi and Ioan Kosztin for sharing the gramicidin simulation data; his work was funded by NIH grant T90 DK070121-04. DCS was supported partially by AFOSR grant FA9550-09-1-0225 and by NSF grants CCF-0634902. RJC's research was supported by a grant from the National Cancer Institute (CA57030) and by Award Number KUS-CI-016-04, made by King Abdullah University of Science and Technology (KAUST). JGM was supported by a postdoctoral training grant from the National Cancer Institute (CA90301)..

†Department of Computational and Applied Mathematics, Rice University, Houston, TX; § Department of Statistics, Texas A&M University, College Station, TX

here has already been applied to modeling experimental AFM trajectories [2, 6, 7].

Statistically, these types of applications can be thought of as semiparametric models that can benefit from regression splines falling under the label of Penalized-splines (P-splines) [8, 9]. We introduce a simple design matrix that simultaneously uses noisy function and derivative scatterplot information to approximate nonlinear curves. The use of both the function and its derivative as "response" data in a P-spline is a unique part of our approach.

The following notation will be used: x_i denotes a design point, $f(x_i)$ represents the function of interest evaluated at x_i, $\partial f(x_i)$ the corresponding derivative $(df(x)/dx|_{x=x_i})$, and ϵ^1, ϵ^2 represent mean zero noise processes, discussed in more detail later, associated with the noisy estimates of $f(x_i)$, $\partial f(x_i)$ respectively. Applications where such data are available include economics [10], geosciences [11], and single-molecule dynamics [2–4, 6, 12–15]. The design matrix constructed for the PuDI method exploits some of the advantageous properties associated with the truncated power function (TPF) basis set [9, 16] and overcomes the well-known ill-conditioning issue associated with this basis by using a recently developed stable and efficient algorithm for computing the penalized least squares solutions associated with the P-spline problem [17]. However other spline basis can be entertained, such as the B-spline basis as advocated when P-splines were introduced by [8]. Smoothing splines can also be considered [18], but the ability of P-splines to parsimoniously represent complex nonlinear functions has appeal in longitudinal data [9] and functional data analysis [19] applications; these techniques show promise in providing a better quantitative understanding of batches of complex single-molecule time series [6, 7, 20]. We demonstrate how information about the (possibly correlated) noise processes can be utilized to improve function estimates using established Generalized Least Squares (GLS) techniques [21] to modify the PuDI design matrix. Illustrative examples demonstrating how undesirable results can be obtained when differences in the noise processes (ϵ^1, ϵ^2) are ignored are presented.

The article is organized as follows: Section 2 quickly reviews established P-splines results [9]. Section 3 presents the basic ideas behind the PuDI method. The background and challenges associated with modeling single-molecule dynamics are presented in Section 4, although we remind the reader that the PuDI method is motivated by single-molecule data sets, it is applicable to other scatterplot situations where derivative information is available (e.g. [10, 11]). Section 5 presents results from both controlled and single-molecule data situations. Section 6 presents the conclusions and outlook. MATLAB scripts for fitting general P-splines with our method are provided in the Supporting Material.

2. Review of P-Spline Notation. The basic regression problem considered here is to approximate a continuous nonlinear function, $f(\cdot)$ using discrete noisy measurements. Most P-spline approximations take the form:

$$y_i = \eta_0 + \eta_1 x_i \dots \eta_p x_i^p + \sum_{j=1}^{K} \zeta_j B_j(x_i), \tag{2.1}$$

where the $B_j(\cdot)$ represent the selected spline basis used. We introduce the following notation: $Z \equiv [B_1, \dots B_K]^\top, u \equiv [\zeta_1, \dots, \zeta_K]^\top, \beta \equiv [\eta_0, \dots, \eta_p]^\top, y \equiv [y_1, \dots y_m]^\top, X_i = [1, \dots, x_i^p], X \equiv [X_1^\top, \dots, X_m^\top]^\top$. With this notation we can rewrite (2.1) as $y = X\beta + Zu = C\beta'$. This mixed model structure serves as a building block for several more complicated semiparametric models [9].

P-splines offer the flexibility of different types of penalties, but we will focus on penalized least squares problems that minimize $\|y - C\beta'\|_2^2 + \alpha\|u\|_2^2$ and use $\widehat{y}$ to denote the least squares estimate for a given α. The data can specify K and/or α. Several methods can be used for selecting such quantities [22]. In this article we use generalized cross validation (GCV) to pick the "optimal" smoothing parameter $\widehat{\alpha} \equiv \text{argmin}_\alpha \text{GCV}(y, \alpha; C)$.

3. The PuDI Method. The PuDI method assumes a noisy unbiased sample of the underlying function and the corresponding derivatives is available. In this situation we write the observed data as

$$z = \{f(x_1), \ldots, f(x_m), \partial f(x_1), \ldots, \partial f(x_m)\}^\top + \epsilon, \tag{3.1}$$

where $\epsilon \equiv (\epsilon_1^1, \ldots, \epsilon_m^1, \epsilon_1^2, \ldots, \epsilon_m^2)^\top$ is assumed normally distributed with mean zero and covariance matrix $\Sigma = WW^\top$. Throughout we assume that Σ is invertible and not poorly conditioned.

The design matrix we propose makes use of the TPF basis, e.g. in Equation (2.1) one sets $B_j(x_i) = (x_i - \kappa_j)_+^p$, where $(\cdot)_+$ is a function that takes real arguments and is the identity for arguments $\geq$ zero and is zero otherwise. We construct the various design matrices using minor transformation of

$$C^{\text{PuDI}} := \begin{pmatrix} 1 & x_1 \ldots x_1^p & (\kappa_1 - x_1)_+^p \ldots (\kappa_K - x_1)_+^p \\ \vdots & \vdots & \vdots \\ 1 & x_m \ldots x_m^p & (\kappa_1 - x_m)_+^p \ldots (\kappa_K - x_m)_+^p \\ 0 & 1 \ldots p x_1^{p-1} & p(\kappa_1 - x_1)_+^{p-1} \ldots p(\kappa_K - x_1)_+^{p-1} \\ \vdots & \vdots & \vdots \\ 0 & 1 \ldots p x_m^{p-1} & p(\kappa_1 - x_m)_+^{p-1} \ldots p(\kappa_K - x_m)_+^{p-1} \end{pmatrix}. \tag{3.2}$$

In terms of a mixed model formulation, the first $p+1$ columns correspond to X, the last K columns correspond to Z, where we assume the random effects $\kappa_i \sim \mathcal{N}(0, \sigma_u^2)$ [9]. With the TPF basis, including of derivative information into the P-spline is straightforward.

We do not claim that the TPF basis is optimal in any sense. However, it can readily handle derivative information estimation in situations where the knot spacing is not uniform, a feature not shared by other popular spline basis like B-splines [8,16]. The intuitive connection to mixed linear models and multivariate regression also has value.

3.1. The Importance of Weighting Derivatives. We next provide a simple illustration demonstrating the importance of weighting measurements of different quality. After we apply simple GLS techniques to C^{PuDI} defined in (3.2), the resulting structure is similar to a system of uncorrelated linear regression equations. The regularization parameter selection, possible departures from normal errors, and nonparametric bias introduced by too small a value of K make the penalized regression spline problem slightly more involved than the example considered, but nonetheless, we demonstrate that intuition on very simple examples carries through to the more complex regression problem. Furthermore the mixed model framework can also be used to get a better understanding of some of the features that make the penalized regression spline problem more involved than standard multivariate regression. For example it is established [9] that when using a TPF basis in a mixed model framework, that the smoothing parameter can be expressed as $\alpha = \sigma_\epsilon^2/\sigma_u^2$, where σ_ϵ^2 represents

the normal variance of the noise associated with the residual regression error and σ_u^2 was introduced in the previous section.

At each design point x_i with $i = 1, ..., m$, we observe a nearly unbiased estimate $z = \{y^{(f)}(x_i), y^{(\partial f)}(x_i)\}$, where $y^{(f)}(x_i)$ is an estimate of the function and $y^{(\partial f)}(x_i)$ is an estimate of the derivative. The PuDI method can be viewed as using two different design matrices to estimate one regression coefficient β. These design matrices are associated with different conditional expectations, i.e. $\mathbb{E}\{y^{(f)}(x)\} = C^{(f)}(x)\beta$ and $\mathbb{E}\{y^{(\partial f)}(x)\} = C^{(\partial f)}(x)\beta$, where $C^{(f)}$(x) and $C^{(\partial f)}(x)$ represent the two distinct design matrices depending on the vector of design points x. Note that the design matrix shown in (3) consists of two matrix blocks, i.e. the block $C^{(f)}(x)$ is stacked on top of $C^{(\partial f)}(x)$. The importance of using generalized least squares can be readily seen with the following simplified multivariate example:

Suppose we are given two sets of independent observations: $(y_1^{(f)}, ..., y_m^{(f)})$ possessing mean μ and variance σ^2 for all x_i and $(y_1^{(\partial f)}, ..., y_m^{(\partial f)})$ having mean μ and variance $c\sigma^2$ for all x_i , where $c > 1$ (c serves as an amplification factor). A possible naive estimate would use only the $y^{(f)}$ data: $\widehat{\mu}_{\text{naive}} = m^{-1} \sum_{i=1}^m y_i^{(f)}$. If the unequal variances are ignored, one might take $\widehat{\mu}_{\text{unweighted}} = (2m)^{-1} \sum_{i=1}^m (y_i^{(f)} + y_i^{(\partial f)})$. The weighted (generalized) least squares estimate would read $\widehat{\mu}_{\text{GLS}} = \{(1 + 1/c)m\}^{-1} \sum_{i=1}^m \left(y_i^{(f)} + (y_i^{(\partial f)}/c)\right)$. All three estimates have mean μ, and the variances in this example are easy to explicitly compute:

$$\begin{aligned} \text{var}(\widehat{\mu}_{\text{naive}}) &= \sigma^2/m; \\ \text{var}(\widehat{\mu}_{\text{unweighted}}) &= \{(1 + c)/4\}\sigma^2/m; \\ \text{var}(\widehat{\mu}_{\text{GLS}}) &= \frac{c}{1 + c}\sigma^2/m. \end{aligned}$$

The variance of the GLS estimate is less than the other two for all c considered (i.e. $c > 1$). If $c > 3$, the unweighted estimate has *larger* variance than the naive estimator. That is, if the estimated derivatives are incorporated without using generalized least squares, one runs the risk of creating a *worse* estimate of β than if one only used the function estimates. For cases where GLS is applied, the results associated with Figure 5.1 provide a demonstration of the improvement obtainable for different c values in a PuDI application; these results show that the simple example above carries through to a more involved setting. The main point of this example was to demonstrate how using an unweighted estimate can do worse than the naive approach. Note also that as $c \to \infty$ the GLS case tends to the naive case so if one estimator is very noisy relative to the other, then the improvement gained by simultaneously using the function and derivative estimation in the P-spline fit diminishes. However, we show even for fairly disparate noise magnitudes a substantial gain can be achieved in various situations.

4. Single-molecule Dynamics. The purpose of this section is to describe single molecule dynamic experiments/simulations and the data that often arise from them. Our approach uses a time series, $\{x_i\}_{i=1}^N$, as input and then applies local maximum likelihood methods (in state-space) to transform the time sequence into a scatterplot sequence of the form $\{\psi_j, f(\psi_j), \partial f(\psi_j)\}_{j=1}^m$ where m is substantially less than the number of temporal observations N. The "ψ_j" can be thought of as local average x value, explained further below. We then apply the PuDI method to this sequence. We demonstrate that incorporating the derivative information substantially improves the model calibrated from observed time series. We stress that this procedure is repeated for *different* time series realizations and substantial variation can be measured

between the curves estimated by P-splines. This variation is due both to the standard uncertainty associated with a finite number of scatterplot observations and due to a latent process, unique to each time series realization, modulating the dynamical response. The variability observed in the curves estimated from different time series is of physical interest and suggests more sophisticated statistical analysis as future work, e.g. functional data regression and/or a longitudinal analysis. This type of "functional" variability has proven to be important to thermodynamic and kinetic computations associated with some single-molecule systems [3, 6, 20].

4.1. Modeling Single-Molecule Dynamics. At time scales currently accessible to many single-molecule experiments, classical statistical mechanics is often assumed to be a highly accurate model of the system dynamics. These models often involve *many* degrees of freedom because each atom possesses a position and momentum vector. Let N_{DOF} denote the number of degrees of freedom. Fairly sophisticated computer simulation algorithms have been developed to include this high level of detail [23]. Let the vector Γ, "the phase space vector", represent all the degrees of freedom of the system. There is interest in determining simplified dynamical summaries, e.g., use $x \in \mathbb{R}^r$ to construct a reduced order model of $\Gamma \in \mathbb{R}^{N_{DOF}}$ with $r << N_{DOF}$. Various motivations exist for appealing to model reduction. The high dimensionality of Γ complicates computer simulations due to the small time step sizes that must be used to ensure numerical stability in numerical integration; a reduced order model can often be simulated for longer length and time scales [24]. In experiments, one can usually only dynamically track a small number of degrees of freedom accurately, but can explore longer length and time scales where various events of scientific interest typically occur [2]. Constructing accurate reduced stochastic dynamical models from time series coming either from single-molecule experiments or computer simulations is one way of comparing these two information sources [2]. Rapid technological advances are closing the time scale gaps in data that can be obtained from experiments and simulations. The smaller time scale gap will facilitate the comparison of models calibrated from experimental and simulation data [23].

Denote the degrees of freedom directly observable, or measured from the simulation, by the vector x. The term "reaction coordinate" is sometimes applied to x-type variables. Some researchers in chemical physics believe that an ideal, or "good", reaction coordinate should be associated with the slowest relevant mode(s) of molecular motion. When this is the case, there is hope for using an effective potential, or potential of mean force denoted here by $\mathcal{U}(x)$, to approximate the dynamics of the high-dimensional system at longer time scales [25]. The approximate force acting on this coordinate is obtained by taking the gradient, $\nabla\mathcal{U}(x)$. In practice it is rarely true that an ideal reaction coordinate is known or measurable from experiments; in these situations one should think of the potential as $\mathcal{U}(x;\Gamma)$ [2]. The unobserved degrees of freedom serve as a latent process and modulate the dynamical response. For simplicity assume that the variable $\chi(\Gamma)$ is a scalar variable that evolves on a time scale much longer than that associated with x. In this situation one should really think of the effective forces as being governed by $\nabla\mathcal{U}(x,\chi)$ and this type of situation is illustrated in Figure 4.1. In what follows we write χ ignoring the Γ dependence, but this lurking variable is implicit whenever χ appears. In molecular modeling, a χ type variable can be one characterizing large scale conformational fluctuations [24]; in a protein this might be a certain dihedral angle. The presence of the slowly evolving latent χ causes physically relevant variability in the P-spline curves we estimate from time series data. The next section outlines more specifically how we transform time

series coming from single-molecule experiments or simulations into scatterplot data.

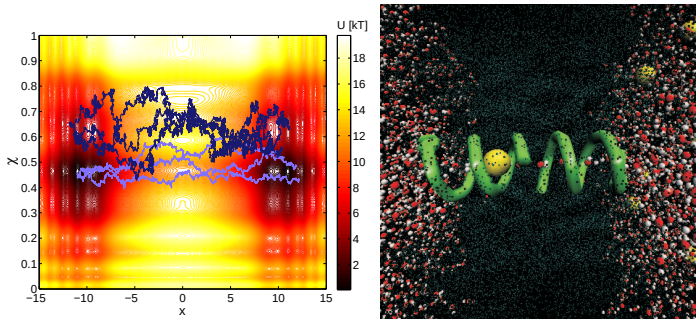

FIG. 4.1. *Contour plot of a fictitious free energy landscape and some sample trajectories (Left Panel). x represents the observable process and the variable χ characterizes the latent process. Three sample trajectories are depicted using two distinct initial values. Two distinct initial conditions are used to stress that randomness is inherent to the reduced dynamics and the initial condition modulates the dynamics (even if the same x value is present at time zero). Since we assume ignorance of the underlying value of χ we would estimate slightly different effective forces (and local diffusion coefficients) due to the ignorance of this information. Snapshot of the gramicidin A channel (Right panel). The helical structure represents a protein complex consisting of two gramicidin A monomers. The large spheres denote potassium ions; the multiple colored spheres denote water molecules and the lightly colored portion represents the lipid bilayer molecules. x corresponds the ion's distance from the channels' plane of symmetry and χ corresponds to a dihedral angle characterizing the complex. Each atom was explicitly modeled (plot generated using the VMD program [26]).*

4.2. Modeling Observable Quantities. Assume batches of time series $\{x_i^{(j)}\}_{i=1}^N$ are collected from the system, where the superscript is used to index the trajectory number and the subscript to index a time ordering of a scalar observable. In the molecular dynamics community, time series are often referred to as "trajectories". For a given trajectory we attempt to fit a stochastic differential equation (SDE) having the form

$$dx_t = \mu(x_t, t; \Gamma)dt + \sqrt{2}\sigma(x_t, \Gamma)dB_t \tag{4.1}$$

where $\mu(\cdot,\cdot)$ and $\sigma^2(\cdot)$ are the nonlinear deterministic drift and diffusion functions (respectively) and B_t represents the standard Brownian motion [27]. We introduce the following terminology "local diffusion coefficient" $\equiv \widetilde{D}(x;\Gamma) := \sigma^2(x;\Gamma)$ in order to distinguish the coefficient in the SDE above from the diffusion coefficient usually implied in the physical sciences: we estimate the former using P-splines. The term "diffusion coefficient" used in statistical physics [28] is not necessarily the same as $\widetilde{D}(x;\Gamma)$. If Γ does not modulate the dynamics, the two definitions are effectively identical.

Diffusion models of the reaction coordinates can be used to approximate a wide range of molecular dynamics simulations [3,20,24]. Unfortunately a parametric model is usually not known *a priori* for the drift and diffusion functions. Other SDE estimation approaches can be entertained [27,29] but, for reasons described more fully in the next section and elsewhere [2,5,6,20], we appeal to local MLE methods. Briefly, the inherent non-stationary nature of the data complicates ones ability to use purely nonparametric methods. The full degrees of freedom vector, Γ, is retained in the drift and diffusion functions to remind us that a latent process is modulating the dynamics. We stress that for *each* observed time series, we estimate a new SDE model.

4.3. From Local MLEs to Global SDEs via P-Splines. The so-called overdamped Langevin equation is a useful approximation in statistical physics [25]. In externally driven systems, the approximation assumes $\mu(x,t) := (k_BT)^{-1}\widetilde{D}(x;\Gamma)\{F(x,\Gamma)+F^{\text{Ext}}(x,t)\}$, where k_BT represents the product of Boltzmann's constant and the system temperature, $F^{\text{Ext}}(x,t)$ denotes the time-dependent force added into the system, and $F(x,\Gamma)$ the effective internal force due to intermolecular interactions. One appeal of overdamped Langevin approximations is that drift and diffusion functions can be physically interpreted, although other SDEs can be entertained. The use of this structure is not necessary, but it does illustrate how our approach is not a traditional nonparametric approach (i.e. it is locally parametric). The goodness-of-fit of the models calibrated from non-stationary data can be checked using the omnibus tests of Hong and Li [30]. These tests have been shown to have adequate power to identify some interesting physical features in molecular dynamics time series [2, 24].

In single-molecule experiments/simulations we often have detailed knowledge of the time-dependent external force added. The use of a local parametric model facilitates incorporating the known time-dependent force added into the single-molecule system in the SDE model. However, we do not know the global functional form of the local force $F(x,\Gamma)$ or local diffusion coefficient $\widetilde{D}(x;\Gamma)$. We use polynomials to model these quantities locally, namely in each neighborhood centered around $\psi_\ell^{(j)}$ the following approximation is used:

$$F(x;\Gamma) = -\frac{\partial}{\partial x}\mathcal{U}(x;\Gamma) \approx A_\ell^{(j)} + B_\ell^{(j)}(x-\psi_\ell^{(j)}); \tag{4.2}$$
$$\sigma(x;\Gamma) = \sqrt{\widetilde{D}(x;\Gamma)} \approx \big(C_\ell^{(j)} + D_\ell^{(j)}(x-\psi_\ell^{(j)})\big),$$

where the local parameter vector, $\theta_\ell^{(j)} \equiv (A_\ell^{(j)}, B_\ell^{(j)}, C_\ell^{(j)}, D_\ell^{(j)})$, is estimated using $\{x_i^{(j)}\}_{i=T_{\ell-1}}^{T_\ell}$. The subscript ℓ is an index of a partition, $1 =: T_0 < \ldots < T_\ell < \ldots < T_m := N$, used to divide the total time series into m disjoint local windows. The results reported were constructed to have ≈ 400 observations fall within a given local window, but "optimal" bandwidth rules in this type of application would be of interest. Also, applying recent local MLE ideas, e.g. [31], to this problem appears to be interesting future research. The constant $\psi_\ell^{(j)}$ denotes a specified point where we wish to evaluate the Taylor type expansion of the expression in Equation (4.2). In this article it corresponds to the temporal average of window ℓ.

The global nonlinear force is constructed by applying the PuDI method to the points $\{\psi_\ell, A_\ell+\epsilon_\ell^1, B_\ell+\epsilon_\ell^2\}_{\ell=1}^m$. The $\sigma(x;\Gamma)$ function is obtained in a similar fashion. The scatterplot data is obtained by finding the $\theta_\ell^{(j)}$ maximizing a likelihood approximation [32] of a local parametric SDE possessing the drift and diffusion given dictated by Equation (4.2) and the external force we add into the system. This is done for m windows. The two functions of interest, $F(\cdot;\Gamma)$ and $\sigma(x;\Gamma)$ have different degrees of smoothness and were estimated independently of one another. We have changed from the generic notation of $\{x_i, f(x_i)+\epsilon_i^1, \partial f(x_i)+\epsilon_i^2\}_{i=1}^m$ used in the introduction to emphasize that the scatterplot information is not directly measured. We have suppressed the superscript indexing the time series number, $^{(j)}$, because each estimated P-spline function constructed in this article uses information from only one time series. The vector $(\epsilon_1^1,\ldots,\ \epsilon_m^1,\ \epsilon_1^2,\ldots,\ \epsilon_m^2)^\top$ is modelled as a normally distributed noise with mean zero and a covariance matrix Σ. This covariance is meant to reflect parameter uncertainty due to finite length of the discrete time series and can be estimated using Monte Carlo simulation of a genuine SDE. More specifically the P-spline curves, i.e.

the drift and diffusion functions, estimated from trajectory j scatterplot data, were used to construct a nonlinear SDE. This SDE was used to simulate multiple new sample paths, here we used 1000, corresponding to trajectory j. We then obtained m sets of local MLE parameters on each simulated path and used this information to approximate the uncertainty in the $\theta_\ell^{(j)}$'s; the empirical covariance between the m vectors was then computed to approximate the parameter uncertainty. This procedure was repeated for each trajectory.

5. Applications. Two sets of applications are studied. The first set of results present Monte Carlo simulation data obtained using discrete samples of known highly oscillatory nonlinear function contaminated with a known normal noise. A relatively small number of scatterplot samples are used to estimate each curve. The intention is to quantitatively study how using derivative information, along with uncertainty estimates, influences the P-spline estimates. The function constructed was meant to mimic the function measured in the ion-channel system of interest which is what we turn to after results on the controlled example are discussed. We then discuss some basic features of the molecular dynamic simulation and present results and discussion associated with this second application. The two sets of results can be read independently, but the information that follows is relevant to both cases.

If the original problem is to find the least squares solution to $C\beta' = y$, then for a given weight matrix $\mathcal{W}$, the GLS analog of this problem would be to find the solution to $\mathcal{W}C\beta' = \mathcal{W}y$. Under the assumption that $\mathcal{W}$ is invertible and not ill-conditioned which is the situation in the cases studied here, the GLS problem can then be viewed as an ordinary least squares problem in a new coordinate system [1], i.e. find the least squares solution to $\tilde{C}\beta' = \tilde{y}$. The penalized least squares problem associated with the P-spline problem requires finding the $\beta' \equiv (\beta, u)^\top$ vector that minimizes $\|\tilde{y} - \tilde{C}\beta'\|_2^2 + \alpha\|u\|_2^2$.

We use several different weight matrices to construct standard least squares problems and these matrices require us to define some parameters: n_{MC} is a parameter determining the number of vectors drawn from a mean zero normal distribution possessing the covariance Σ. These vectors are used to form a simple empirical estimate of Σ. Note that in the first application the Σ associated with the scatterplot data is known and used to generate the Monte Carlo samples, in the second application we assume that the parameter distribution of the local MLE procedure can be adequately approximated by a normal distribution although the associated covariance is *not* known to us in closed-form; the procedure we use in this application was described earlier. The various weight and design matrices considered are listed below.

Case 1 $C_1 = W_1^{-1}C^{\mathrm{PuDI}}$, and W_1 is the Cholesky factor of the measurement noise covariance: recall, that this is known exactly in the first benchmark application.

Case 2 $C_2 = W_2^{-1}C^{\mathrm{PuDI}}$, and W_2 is the Cholesky factor of the estimated measurement noise covariance using $n_{MC} = 5 \times 10^4$.

Case 3 $C_3 = W_3^{-1}C^{\mathrm{PuDI}}$, and W_3 is the Cholesky factor of the estimated measurement noise covariance using $n_{MC} = 1 \times 10^3$.

Case 4 $C_4 = C^{\mathrm{PuDI}}$.

Case 5 $C_5 = P_1C^{\mathrm{PuDI}}$, where $P_1 = (I_{m\times m}, 0_{m\times m})$ i.e., use only function information.

Case 6 $C_6 = P_2C^{\mathrm{PuDI}}$, where $P_2 = (0_{m\times m}, I_{m\times m})$ i.e., use only derivative information.

[1] If Σ happens to be ill-conditioned, numerical methods exist for treating this situation [21].

The C^{PuDI}, design matrix was formed using the TPF basis parameters $K = 20$ and $p = 2$. The quantiles were used to select the knot locations [33, 34]. The regularization/smoothing parameter was selected using GCV in all cases. The results did not change appreciably if we used other criteria, e.g. AIC, used $p = 3$ or if we increased K. [9].

5.1. Benchmarking PuDI on a Smooth Nonlinear Function. Here we quantitatively study how a PuDI type design matrix can assist in the estimation of a known function under different noise conditions on the scatterplot data. Recall this function was constructed to mimic the salient features of curves coming from single-molecule data studied later. This function is written explicitly in the Appendix.

5.1.1. Data generation. Each grid point x_i was associated with the noisy measurements $f(x_i) + \epsilon_i^1$ and $\partial f(x_i) + \epsilon_i^2$. An i.i.d. two-dimensional Gaussian noise with mean zero and covariance matrix

$$\tilde{\Sigma} \equiv \begin{pmatrix} \sigma_f^2 & \rho\sigma_f\sigma_{\partial f} \\ \rho\sigma_f\sigma_{\partial f} & \sigma_{\partial f}^2 \end{pmatrix}$$

was used to generate the noise for each grid point x_i. The diagonal of this 2×2 matrix was varied, one diagonal component was always set to be unity, and the correlation coefficient, ρ, between each ϵ_i^1 and ϵ_i^2 was set to zero in the plot shown in this subsection. The net covariance matrix, Σ, associated with the P-spline scatterplot data was sparse due to the i.i.d. noise structure used. In the $\rho = 0$ case, Σ is a diagonal matrix defined by the vector $(\sigma_f^2, \ldots, \sigma_f^2, \sigma_{\partial f}^2, \ldots, \sigma_{\partial f}^2)^\top \in \mathbb{R}^{2m}$. The $\rho > 0$ case covariance had the same structure plus two off-diagonal bands each consisting of m repeated entries of the product $\rho\sigma_f\sigma_{\partial f}$. Tables reporting results with $\rho = 0.5$ and $\rho = 0$ are reported in the Supplementary Material; the same qualitative trends are observed in each case.

5.1.2. Results and Discussion. Figure 5.1 plots the logarithm of the average average mean square error (AMSE) associated with predicted f and ∂f for various Σ's. The PuDI estimates using the weights, design matrices C_1-C_3, outperform all other methods. In both f and ∂f, as the ratio of the diagonal terms of Σ tend to 0 or ∞, the PuDI estimate approaches that of the "naive estimator". The naive estimator uses design matrix C_5 or C_6, with the selection depending on $\sigma_f/\sigma_{\partial f}$ as shown in Figure 5.1. The limits of 0 or ∞ mentioned above indicate the extra information provided by using both f and ∂f is negligible in relation to the information accessible to the naive estimator. However there is significant gain for a large range of $\sigma_f/\sigma_{\partial f}$ values. Also note that using an empirical covariance approximation was nearly identical to the case where the exact covariance Σ was used. Note also that the estimate of the Cholesky factor, obtained using n_{MC} sample vectors of the Σ matrix was dense whereas the known underlying Cholesky factor, W was highly sparse. The sampling noise caused the estimated Cholesky factor to appear dense, but this artifact did not hurt the PuDI estimate utilizing W_2 or W_3.

The vertical lines denote the point where the average AMSE estimator using both f and ∂f but ignoring the different noise variances, i.e. use design matrix C_4, is greater than that of applying the naive estimator to the less noisy random vector. The intuition gained from the two dimensional normal variable result dictates that this cross-over should occur when the noisier estimate is a factor of three greater than the other estimate. Recall that our situation is more involved due to the smoothing

parameter selection and other tunable parameters, but nonetheless the crossover occurs close to three. The exact crossover point depended on whether f was noisier than ∂f or vice versa. Note that in the PuDI design matrix cases, we utilized estimates of $f(x_i)$ and $\partial f(x_i)$ and selected $\widehat{\alpha}$ which minimized the GCV consistent with both measurements. Since the smoothness is different in each function, the resulting $\widehat{\alpha}$ represents a type of weighted average between the $\widehat{\alpha}$ that would have been selected had only $f(x_i)$ or $\partial f(x_i)$ been used individually. The mixed model formulation allows one to readily see why for a GLS estimate, that the smoother function dominates the α selection.

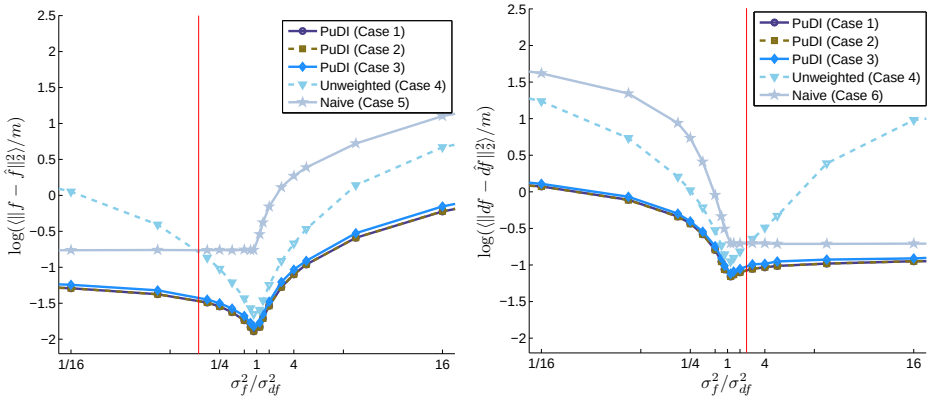

FIG. 5.1. *AMSE of $f(x)$ (left panel) and $\partial f(x)$ (right panel) using various semiparametric estimators. The x-axis plots the ratio of the variance of the f noise to that of the ∂f noise and the y-axis contains the AMSE measured over 1×10^4 Monte Carlo simulations. In each case the weighted PuDI methods (Cases 1-3) outperform the other estimators. These plots also demonstrate how findings from simple multivariate arguments carry through to these nonlinear semiparametric regression spline fits. The vertical red lines indicate the point at which the naive estimator (using only f or ∂f information) outperforms the PuDI estimator using f and ∂f, but weighting the observations equally (i.e. Case 4). See text for additional details.*

5.2. Applying PuDI to Estimate SDEs Characterizing Ion-Channel Dynamics. For those not interested in the fine details, one can think of this application as a type of study in longitudinal data analysis [9]. There are several subject specific responses and the deviations from the mean population function provides useful information about the individual curves, which here correspond to unobserved, but physically important, phase space information. The interest is in the different types of effective forces experienced by a potassium ion as it travels across a pore formed by a single protein lodged in a lipid bilayer. This lipid bilayer serves as a boundary between the interior and exterior of a cell and does not permit water or ions to easily pass in the absence of an open ion channel. A schematic of the gramicidin A ion channel studied is provided in Fig. 4.1. This particular system was selected because it has been extensively studied both experimentally and theoretically. This ion-channel is commonly used as a benchmark in molecular dynamics simulations [35]. The results we study introduce an external force into the system to "steer" an ion across the channel in a prescribed time. Measurements from these simulations can and be used to back out a potential of mean force and diffusion coefficient using recently developed nonequilibrium statistical mechanics methods. These quantities are often of interest in a variety of single-molecule simulations. We demonstrate how capturing

the variation induced by χ type variables is important for making predictions. The physical relevance of this type of variability is described in detail elsewhere [3].

5.2.1. Data generation. The NAMD program [36] was used to generate steered molecular dynamic simulations [25] consisting of $36,727$ atoms. Constant particle number, pressure and temperature (NpT) simulations were used. The x coordinate corresponds to the distance between the center of mass of the channel and the ion's axial location within the channel; this position was recorded every $0.1\ ps$ for $1\ ns$. The resulting time series were then divided into $m = 40$ disjoint windows and the P-spline data was obtained from the sequence of local MLEs taken along this partition. In all cases, the estimated local MLE parameters along with the in sample time series passed goodness-of-fit tests appropriate for the non-stationary data [30]. A more detailed account of the simulation methodology is reported in [3].

5.2.2. Results and Discussion. Figure 5.2 displays the global nonlinear effective force obtained using 10 separate steered molecular dynamics realizations. Only results obtained using design matrices C_2,C_4, and C_5 were considered because the true Σ is unknown and the interest is in the function itself (not the derivative). We observe that results obtained using C_2,C_4 appear roughly similar, but C_5 appears to be over smoothing due to the lack of derivative information. The rightmost panel focuses on the binding pocket of the channel. This binding pocket is a local minimum on the free energy landscape; here we see that the differences between the C_2 and C_4 curves are more pronounced.

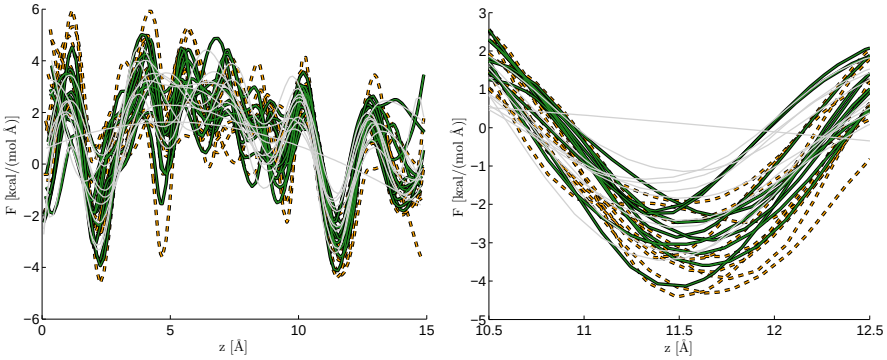

FIG. 5.2. *The different curves correspond to the effective force estimated from 10 realizations of time series data. After processing these time series, we obtained sets of 10 scatterplot data. These data sets were then processed with design matrices C_2 (dark solid), C_4 (dashed), and C_5 (light grey). The underlying scatterplot data is the same regardless of the design matrix used; differences in curves are due only to the P-spline design matrix. See text for additional information.*

Once the P-spline is estimated, we can construct a global nonlinear SDE (see Sec. 4.3) and then simulate multiple realizations of the process using a large number of Brownian paths. The multiple Brownian paths are supposed to quantify the inherent variability caused by neglecting "uninteresting" fast-scale motion in the detailed dynamics. This type of variability, associated with one steered molecular dynamics path, can be important in several contexts [2, 5].

The nonequilibrium work associated with steered molecular dynamics simulations is one example illustrating the item above. The work tubes associated with a single steered molecular dynamics realization can be computed using the estimated SDEs.

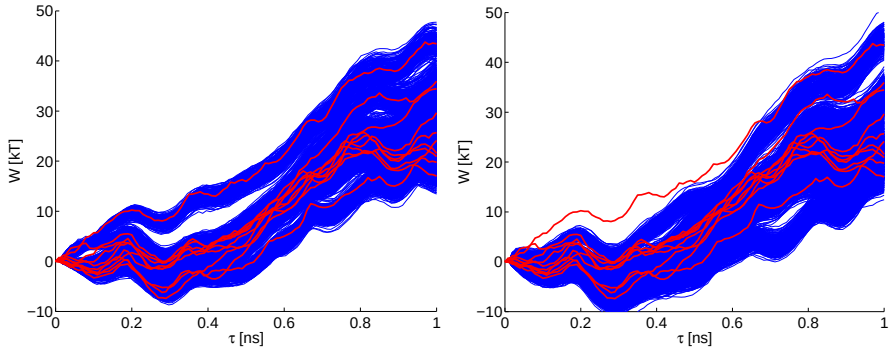

FIG. 5.3. *The work tubes simulated using the 10 SDEs constructed from stitching together local SDE models by P-spline design matrices C_2 (left panel) and C_4 (right panel). Each work tube is made of 1000 work trajectory simulations using the 10 global SDEs previously mentioned. The thick lighter color curves correspond to the work trajectories measured directly from the 10 steered molecular dynamics simulation (one work trajectory per steered molecular dynamics trajectory). The P-splines used the same scatterplot data in the left and right panel, only the design matrix changes and this alone explains the difference in the estimated curves.*

We can use the SDEs to simulate the nonequilibrium work and compare the variability *between* these work tubes. Each tube is computed from information contained in one SDE corresponding to one molecular dynamics trajectory. The variability between tubes provides information about the fluctuations induced by a latent χ type process whereas the width of each single tube can be attributed to fast-scale noise experienced by the steered molecular dynamics path. Figure 5.3 demonstrates that the different level of smoothing associated with C_2 and C_4 substantially affects the predictive ability of the corresponding global SDE model. Note that the underlying scatterplot points are the same in all cases, only the P-spline design matrix changes. The different predictions have consequences in physical quantities computed from these simulated work paths. For example using C_2 gives improved potential of mean force and diffusion coefficient estimates compared to other methods [3].

6. Conclusions and Outlook. We demonstrated how a single-molecule time series can be transformed, via local maximum likelihood type methods, into scatterplot data approximating pointwise function and derivative information associated with a SDE. The functions needed by a SDE approximating the global dynamics of the time series were obtained using P-spline techniques. The PuDI design matrix was shown to be useful in this context. The PuDI design matrix exploited some of the advantageous properties of the TPS basis; numerical difficulties were overcome with a recent algorithm [17]. The use of GLS along with P-splines was shown to influence the estimated curves and the difference was shown to be relevant in regards to simulating physical quantities of interest. For example the work computation associated with the ion-channel system studied benefited substantially from the GLS implementation. When this procedure was repeated for different time series, it was shown that the global SDE functions estimated from different time series exhibited variation in part due to a latent process, i.e. our data consisted of "subject specific curves". We briefly discussed why this is relevant information to modern biophysics applications [15, 24].

Although we focused on simulation data, the methodology is also applicable to experimental data [2, 6, 7]. Applications making fuller use of pointwise function estimates and derivative proxies calibrated from time series, as the PuDI method was

demonstrated to do, show promise as tools that can be used for understanding the rich amount of information contained in recent single-molecule experiments and computer simulations. Other areas where function and derivative scatterplot information is available and a PuDI might be helpful include geosciences [11] and finance [10]. MATLAB scripts illustrating the PuDI method are available via the Supporting Materials.

REFERENCES

[1] S. Kou. Stochastic modeling in nanoscale biophysics: subdiffusion within proteins. *Annals of Applied Statistics*, 2:501–535, 2008.

[2] C.P. Calderon, N. Harris, C.-H. Kiang, and D.D. Cox. Quantifying multiscale noise sources in single-molecule time series via pathwise statistical inference procedures. *J. Phys. Chem. B*, 113:138, 2009.

[3] C.P. Calderon, L. Janosi, and I. Kosztin. Using stochastic models calibrated from nanosecond nonequilibrium simulations to approximate mesoscale information. *J. Chem. Phys.*, 130:144908, 2009.

[4] J. P. Junker, F. Ziegler, and M. Rief. Ligand-dependent equilibrium fluctuations of single calmodulin molecules. *Science*, 323:633–637, 2009.

[5] C.P. Calderon. On the use of local diffusion for path ensemble averaging in potential of mean force computations. *J. Chem. Phys.*, 126:084106, 2007.

[6] C.P. Calderon, W.-H. Chen, N. Harris, K.J. Lin, and C.-H. Kiang. Analyzing DNA melting transitions using single-molecule force spectrscopy and diffusion models. *J. Physics: Condensed Matter*, 21:034114, 2009.

[7] C. P. Calderon, N. C. Harris, C. H. Kiang, and D. D. Cox. Analyzing single-molecule manipulation experiments. *J. Mol. Recognit.*, May 2009.

[8] P.H.C. Eilers and B.D. Marx. Flexible smoothing with B-splines and penalties (with discussion). *Statistical Science*, 11:89–121, 1996.

[9] D. Ruppert, M.P. Wand, and R.J. Carroll. *Semiparametric Regression*. Cambridge University Press, New York, 2003.

[10] P. Hall and A. Yatchew. Nonparametric estimation when data on derivatives are available. *Ann. Statist.*, 35:300–323, 2007.

[11] D.D. Cox. Approximation of method of regularization estimators. *Ann. Statist.*, 16:694–712, 1990.

[12] S. B. Smith, Y. Cui, and C. Bustamante. Overstretching B-DNA: The elastic response of individual double-stranded and single-stranded DNA molecules. *Science*, 271:795 – 799, 1996.

[13] H. Clausen-Schaumann, M. Rief, and H. E. Gaub. Sequence-dependent mechanics of single DNA molecules. *Nat. Struct. Biol.*, 6:346–349, 1999.

[14] H. Lu, B.Isralewitz, A. Krammer, V. Vogel, and K. Schulten. Unfolding of titin immunoglobulin domains by steered molecular dynamic simulations. *Biophys. J.*, 75:662671, 1998.

[15] D. N. Fuller, D. M. Raymer, V. I. Kottadiel, V. B. Rao, and D. E. Smith. Single phage T4 DNA packaging motors exhibit large force generation, high velocity, and dynamic variability. *Proc. Natl. Acad. Sci. U.S.A.*, 104:16868–16873, Oct 2007.

[16] C. de Boor. *A practical guide to splines*. Springer, New York, 2001.

[17] C.P. Calderon, J.G. Martinez, R.J. Carroll, and D.C. Sorensen. Psqr: A stable and efficient penalized spline algorithm. *submitted*, 2009.

[18] K. V. Mardia, J. T. Kent, C. R. Goodall, and J. A. Little. Kriging and splines with derivative information. *Biometrika*, 83(1):207–221, 1996.

[19] J. Ramsay and B.W. Silverman. *Functional Data Analysis*. Springer-Verlag, New York, 2005.

[20] C.P. Calderon and R. Chelli. Approximating nonequilibrium processes using a collection of surrogate diffusion models. *J. Chem. Phys.*, 128:145103, 2008.

[21] G.H. Golub and C.F. van Loan. *Matrix Computations*. Johns Hopkins University Press, Balitimore, MD, 1996.

[22] D. Ruppert and R. J. Carroll. Spatially adaptive penalties for spline fitting. *Australia and New Zealand Journal of Statistics*, 42:205–223, 2000.

[23] M. Sotomayor and K. Schulten. Single-molecule experiments in vitro and in silico. *Science*, 316:1144 – 1148, 2007.

[24] C.P. Calderon and K. Arora. Extracting kinetic and stationary distribution information from short md trajectories via a collection of surrogate diffusion models. *J. Chem. Theory &*

Comput., 5:47, 2009.
[25] S. Park and K. Schulten. Calculating potentials of mean force from steered molecular dynamics simulations. *J. Chem. Phys.*, 120:5946–5961, 2004.
[26] William Humphrey, Andrew Dalke, and Klaus Schulten. VMD – Visual Molecular Dynamics. *Journal of Molecular Graphics*, 14:33–38, 1996.
[27] B.L.S. Prakasa Rao. *Statistical Inference for Diffusion Type Processes*. Hodder Arnold, 1999.
[28] R. Zwanzig. *Nonequilibrium Statistical Mechanics*. Oxford University Press, New York, 2001.
[29] Y. Pokern, A.M. Stuart, and E. Vanden-Eijnden. Remarks on drift estimation for diffusion processes. *Multiscale Model. Simul.*, in press, 2009.
[30] Y. Hong and H. Li. Nonparametric specification testing for continuous-time models with applications to term structure of interest rates. *The Review of Financial Studies*, 18:37–84, 2005.
[31] J. Fan, Y. Fan, and J. Jiang. Dynamic integration of time- and state-domain methods for volatility estimation. *Journal of American Statistical Association*, 102:618–631, 2007.
[32] J.C. Jimenez and T. Ozaki. An approximate innovation method for the estimation of diffusion processes from discrete data. *J. Time Series Analysis*, 27:77–97, 2006.
[33] D. Ruppert. Selecting the Number of Knots for Penalized Splines . *J. Computational & Graphical Statistics*, 11:735–757, 2002.
[34] V. Baladandayuthapani, B. K. Mallick, and R.J. Carroll. Spatially Adaptive Bayesian Penalized Regression Splines (P-splines). *J. Comp. Graph. Stat.*, 14:378–394, 2005.
[35] T. W. Allen, T. Bastug, S. Kuyucak, and S. H. Chung. Gramicidin A channel as a test ground for molecular dynamics force fields. *Biophys. J.*, 84:2159–2168, Apr 2003.
[36] J. Phillips, R. Braun, W. Wang, J. Gumbart, E. Tajkhorshid, E. Villa, C. Chipot, R. D. Skeel, L. Kale, and K. Schulten. Scalable molecular dynamics with namd. *J. of Comp. Chem.*, 26:1781–1802, 2005.

7. Supplementary Material. Reference function (studied in Section 5.1) constructed to mimic features of effective forces in the ion-channel modeled was

$$y = -\left(2.5\sin(2\pi\frac{6x}{15}) + 9\exp(-\frac{(x-3)^2}{2}) + 5\exp(-\frac{(x-10)^2}{2}) - 2\right)/3; \tag{7.1}$$

Note: the channel is symmetric with respect to $x = 0$. We studied the situation where $x \in [0, 15]$ whereas the expression above is valid for $x \in [-15, 0]$ (the net channel spans -15 to 15 in units of Å [3]). A simple reflection at x=0 gives the result we report.

TABLE 7.1

Average AMSE $\langle \|f - \widehat{f}\|_2^2/m \rangle$ and measured standard deviation in parentheses for several estimators of f in the MC study described in Section 5.1. The subscript corresponds to the design matrix used and the "Noise Ratio" refers to the ratio of σ_f^2/σ_{df}^2. The correlation coefficient ρ was set here to 0.5 (this parameter is needed to specify Σ).

Noise Ratio	f_1	f_2	f_3	f_4	f_5
1/16	0.28(0.10)	0.27(0.10)	0.29(0.10)	1.05(0.36)	0.47(0.18)
1/4	0.21(0.08)	0.21(0.08)	0.22(0.08)	0.36(0.13)	0.47(0.18)
1/2	0.18(0.07)	0.18(0.07)	0.19(0.07)	0.24(0.09)	0.47(0.19)
2	0.22(0.11)	0.22(0.11)	0.23(0.11)	0.29(0.14)	0.86(0.26)
4	0.33(0.19)	0.33(0.19)	0.36(0.20)	0.51(0.26)	1.31(0.46)
16	0.80(0.63)	0.80(0.63)	0.86(0.68)	1.95(0.92)	3.00(2.06)

TABLE 7.2

Average AMSE $\langle \|f - \widehat{f}\|_2^2/m \rangle$ and measured standard deviation in parentheses for several estimators of f in the MC study described in Section 5.1. The subscript corresponds to the design matrix used and the "Noise Ratio" refers to the ratio of σ_f^2/σ_{df}^2. The correlation coefficient ρ was set here to 0 (this parameter is needed to specify Σ).

Noise Ratio	f_1	f_2	f_3	f_4	f_5
1/16	0.36(0.14)	0.36(0.14)	0.39(0.15)	1.06(0.36)	0.47(0.18)
1/4	0.27(0.10)	0.27(0.10)	0.29(0.10)	0.36(0.13)	0.46(0.18)
1/2	0.22(0.09)	0.22(0.09)	0.24(0.09)	0.24(0.09)	0.47(0.18)
2	0.27(0.13)	0.27(0.13)	0.29(0.14)	0.29(0.13)	0.86(0.26)
4	0.41(0.22)	0.41(0.22)	0.43(0.24)	0.51(0.25)	1.30(0.44)
16	0.93(0.69)	0.93(0.69)	0.98(0.72)	1.94(0.91)	2.98(2.03)